Google   Visualizza questa pagina in: Italiano   **Traduci**   Disattiva per: Inglese   Opzioni ▼

G+1 20   Altro   Blog successivo»   cabasino@gmail.com   Bacheca   Esci

Arduino Basics

An Arduino tutorial blog. Free Arduino tutorials for everyone !

| Home | Arduino Basics Projects Page | Forum | Contact Author | Donate |

20 December 2013

PIR Sensor (Part 2)

In this tutorial we will connect our HC-SR501 PIR (movement) Sensor to an Arduino UNO. The PIR sensor will be powered by the Arduino and when movement is detected, the PIR sensor will send a signal to Digital Pin 2. The Arduino will respond to this signal by illuminating the LED attached to Pin 13.

PIR Sensor (Part 1) : Showed that this sensor can be used in isolation (without an Arduino). However, I will still demonstrate how you can attach this sensor to the Arduino so that we can move forward to more advanced objectives and concepts.

Video

ArduinoBasics: HC-SR501 PIR Tutorial 2

▶

Parts Required

- Mini Breadboard 4.5cm x 3.5cm
- PIR Motion sensor (HC-SR501)
- LED and 330ohm resistor

Translate

Seleziona lingua

Powered by Google Traduttore

Search This Blog

[   ]   Search

Pages

Arduino Basics Projects Page

Arduino Basics YouTube Videos

Arduino Webserver Data Viewer

Donate

Subscribe (Feed)
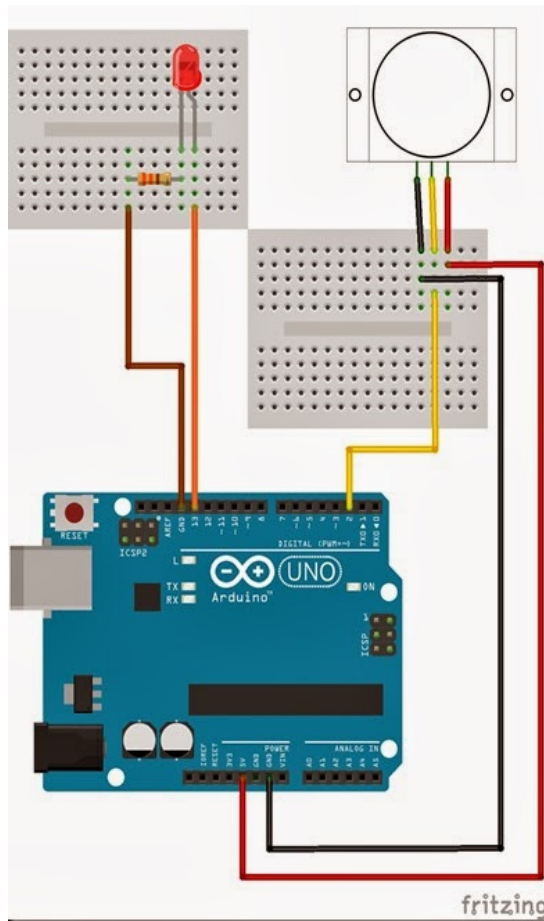
Subscribe

RSS Feed

Total Pageviews

3,672,747

- Arduino UNO (or compatible board)

Fritzing Sketch

Arduino Sketch

```
 1 /*Simple PIR sketch: Written by ScottC, 19th Dec 2013
 2
 3  http://arduinobasics.blogspot.com/
 4
 5  -------------------------------------------------*/
 6
 7 void setup(){
 8  pinMode(13,OUTPUT);
 9  pinMode(2,INPUT);
10 }
11
12 void loop(){
13  digitalWrite(13,digitalRead(2));
14 }
```

The sketch above reads the signal coming in from the PIR sensor on Pin 2, and if it reads HIGH, it light up the LED attached to Pin 13. If it reads LOW, it will turn the LED off. This is all controlled by line 13 in the Arduino

The following table helps to identify the purpose of the potentiometers on the PIR sensor. Most people say they use trial and error. I will attempt to reduce the mystery of these components on the PIR board.

104 (Left) – Max

LED on = 20 sec
LED off = 3 sec

When you move the 104 labelled potentiometer all the way to the left (max position), the LED will remain on for 20 seconds after movement is detected. The 20 seconds is independent of the other potentiometer (105) setting. When the LED turns off, it will remain off for 3 seconds before the sensor will trigger again from any further movement.

104 (Right) – Min

LED on = 1 sec
LED off = 3 sec

When you move the 104 labelled potentiometer all the way to the right (min position), the LED will remain on for 1 second after movement is detected. When the LED turns off, it will remain off for 3 seconds before the sensor will trigger again from any further movement.

105 (Left) – Max

Most sensitive – Detects movement from over 10 steps away.

The 105 labelled potentiometer controls the sensitivity of the PIR sensor. When in the left position, the PIR sensor is most sensitive and small amounts of movement will trigger the sensor. It detected my movement (ie a single step to the left of right) from over 10 steps away from the sensor. I was very impressed.

105 (Right) – Min

significant movement for sensor to trigger. Detects movement from about 4 steps away.

When the 105 labelled potentiometer is twisted to the right, the PIR sensor becomes a lot less sensitive. I needed to take much bigger steps to the left or right to trigger the sensor (about 3 times the size compared to the left position). It also had a lot more trouble detecting movement occurring further away. It only really started to detect my movement when I was about 4 steps away from it.

My preferred combination was 104-Right (min) + 105-Left (max), which meant that the sensor would remain on for only a short period of time, and detect any subtle movements in the room. This combination is displayed below:
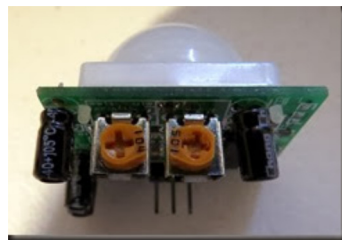
I have not tested to see how it performs over a very long period with this setting, and whether it would suffer from false positive readings, but that could easily be fixed by turning the 105 labelled potentiometer a bit more to the right.

Labels: Arduino, ArduinoBasics, best arduino blog, HC-SR501, LED, movement, PIR, project, Sensor, tutorial

28 comments:

Anonymous 18 January 2014 at 15:58

This comment has been removed by a blog administrator.

Reply

> Replies
>
> **Scott C**   19 January 2014 at 11:15
>
> May plot some sensors soon using MegunoLink . Looks interesting.

Reply

Anonymous 22 February 2014 at 17:04

Thank you for explanation.

Reply

ibra hasan 19 April 2014 at 06:33

i have problem, this range not to far, just 10CM, the sensitivity is max. please help me.

Reply

> Replies
>
> **Scott C**   24 April 2014 at 08:52
>
> Are you using the same PIR sensor that I am using?

Reply

Anonymous 27 April 2014 at 21:12

Hi. I just seem to get a regular output of so many Motion detects to no motion detected. Its like its in a loop of 5 high outputs to a bout 20 lows that just repeats? Do you think my sensor is faulty?

Reply

Replies

Scott C　　29 April 2014 at 19:21

Sounds very similar to a faulty sensor that I had.

Reply

Anonymous 28 July 2015 at 05:32

Hi !
I want to connect my PIR sensor to light up a 12v LED lamp whenever a person is detected in a room. Is it safe to connect the lamp directly to my arduino ?

Reply

Replies

Scott C　　28 July 2015 at 09:08

No - not safe.
You will need to power the LED lamp with an alternate power supply. The Arduino could control the LED lamp switch using a Relay or MOSFET.... or you may be able to get away with a transistor. It just depends on how much power the LED lamp consumes... You would need to find out which option is best for your project.

Reply

Oussama Ghouagh 27 October 2015 at 19:58

Hi !
I'm working in arduino project which will keep an eye on my little dog while I'm not home and send me an alert if he makes an unusual movement. Can I do that with a PIR sensor based on comparing the frequencies it sends ?
Thank you

Reply

Replies

Scott C　　27 October 2015 at 21:16

The sensor that I used, "triggered" when movement was detected. You can adjust the sensitivity of the sensor (as discussed in the tutorial).... but there is nothing really "smart" about this module... ie. it cannot tell whether the movement is "normal" or "unusual". But I guess you can build some smarts within your Arduino program, but you are limited to movement or no movement. By using a number of these sensors, you may be able to identify if the movement is "unusual" - but that is a very vague statement, because I don't really understand what you qualify as unusual.

Reply

Oussama Ghouagh 28 October 2015 at 00:06

I mean by unusual, any movement that gives a warning signal, for example : being attacked by something : snake maybe since the dog's house is outdoor or any thing else

Reply

Replies

Scott C　　28 October 2015 at 12:03

It won't really indicate the amount of movement - only that it moved.

I am not sure if it would detect a snake - because it is cold blooded??

If it were me trying to do what you want to do - I would use some sort of web-cam setup.

Oussama Ghouagh 29 October 2015 at 18:33

Ok, that's a good idea, with a web-cam I think I could do that.
Thank you !

Scott C　　　3 November 2015 at 20:22

Something like this may be what you are looking for :
Original Xiaomi Small Ants intelligent HD Camcorder Network Wireless Camera

Reply

Unknown 8 January 2016 at 20:08

My arduino uno goes off when I touch the wire attached to pin 2.This is with no PIR attached to the uno. Now it is going off when I get near the uno itself. Any ideas?

Reply

　Replies

Scott C　　　10 January 2016 at 00:12

Floating pin. You need to attach the PIR sensor (while the Arduino is not powered).

Although I should check, when you say "goes off" - do you mean the LED turns on ? Or do you mean the Arduino turns off ?

Reply

Anonymous 11 February 2016 at 16:53

Hi
I made the same as connection you have described, but it doesnt work. I have these cheap items from china :
http://www.ebay.com/itm/251733271073?_trksid=p2060353.m2749.l2649&
ssPageName=STRK%3AMEBIDX%3AIT
http://www.ebay.com/itm/151919077902?_trksid=p2060353.m2749.l2649&
ssPageName=STRK%3AMEBIDX%3AIT

Could it be bad item or where is problem?
Thanks

Reply

　Replies

Scott C　　　11 February 2016 at 19:38

Does it work when you carry out the steps in part 1:
http://arduinobasics.blogspot.com.au/2013/12/pir-sensor-part-1.html

Anonymous 11 February 2016 at 23:24

I dont know why, but after reconnection from usb it started work... I hope it will work now.
Is possible to change 3 sec LED off some way?

Reply

Anonymous 11 February 2016 at 22:59

I just tried part 1 and it works correctly... when I connect arduino and load sketch it doesnt work and its only blinking led...

Reply

Anonymous 10 May 2016 at 19:20

When the LED turns off, how to make the sensor remain off only for 1 seconds before the sensor trigger again from any further movement?

Reply

Unknown 7 June 2016 at 05:17

Nice tutorial. Thanks!

Reply

Anonymous 8 June 2016 at 12:34

do you know how to use this sensor with other sensor? for example ear-clip heart rate sensor

Reply

Replies

Scott C   9 June 2016 at 00:04

Yes - you would have to read them on a separate digital pin from each other. So you would leave one attached to Digital pin 2, and have one of the others attached to another digital pin like D4.
Then take a reading as per normal.

Reply

Andharhe Hutabarat 9 June 2016 at 22:57

hi scott. nice tutorial.
btw, how to keep the LED on as long as there's motion detected?

Reply

Replies

Scott C   10 June 2016 at 15:30

Set the 104 potentiometer to the far right. With the code above, this will turn the LED on for 1 second and off for 3 seconds. So the detection cycle (when there is constant movement) is about once every 4 seconds.

So instead of illuminating the LED based solely on the PIR sensor's output, you would illuminate the LED based on a timer within the Arduino code.

Poll digital pin 2, and every time you detect movement, you reset a count-down timer.
The count-down timer must be set to a timeframe greater than 4 seconds. And if you detect movement before the time runs out, you reset the timer (to 6 seconds or whatever you want)and keep the LED on until the timer runs out. When the timer runs down to zero seconds you turn off the LED... but it will never reach zero if motion is detected. Does that make sense?

Reply

raj kumar 30 August 2016 at 00:26

sir iam new to arduino board any step by step basic file.

Reply

```
Enter your comment...
```

Comment as: Simone Cabasino (**Sign out**

Publish   **Preview**   ☐ Notify me

Feel free to leave a comment about this tutorial below.
Any questions about your particular project should be asked in the ArduinoBasics forum.

Comments are moderated due to large amount of spam.

Subscribe to: Post Comments (Atom)

G+1   7.7k

Featured Solution: Cayenne with Arduino

Cayenne is a clever Arduino IoT solution, which allows you to build your project using a very easy to use drag-and-drop interface. Very little (if any) programming is required. You can create your very own internet dashboard to monitor your sensors (from your browser or phone app), or if you want, you can use the interface to control a motor, turn on a light, unlock a door. Test it out yourself - by joining the early beta. All you have to do is click on the picture below and sign up. It's free !
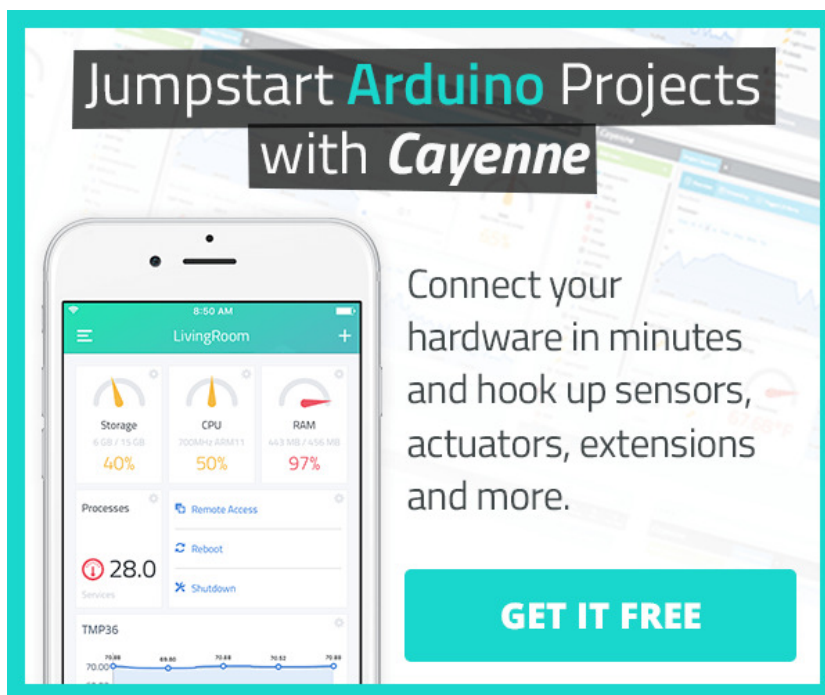
This Week's Most Popular Posts

433 MHz RF module with Arduino Tutorial 1

Relay Module

HC-SR04 Ultrasonic Sensor

Simple Arduino Serial Communication

433 MHz RF module with Arduino Tutorial 2

Get Email Notifications of My Latest Posts

Email address...   Submit

Subscribe To ArduinoBasics

🔊 Posts   ⌄

🔊 Comments   ⌄

Awesome Inc. template. Powered by Blogger.